

PHP



المركز الرئيسي : 11شارع د/ محمد رأفت – محطة الرمل – الإسكندرية

تليفون وفاكس : (+2)(03) 4838326

موبايل : (+2) 01001634294

www.daralbraa.com

Email: info@darAlbraa.com

أسامة محمد فتحي

Osama-muhammed@hotmail.com



الناشر:	**دار البراء لنشر وتوزيع الكتب العلمية**
رئيس مجلس الإدارة:	**إبراهيم محمد إبراهيم زبير**
اسم الكتاب:	لغة PHP
المؤلف:	أسامة محمد فتحي
رقم الإيداع:	
الترقيم الدولي:	
المقاس:	14.8 X 21
عدد الصفحات:	68
العنوان:	11 شارع د/محمد رأفت – محطة الرمل – الإسكندرية
تليفون وفاكس:	4838326 (03)(2+)
للاستعلام والمبيعات:	01001634294 (2+)
البريد الالكتروني:	**info@daralbraa.com**
الموقع:	WWW.DarAlbraa.com

تركيب بي اتش بي ..

الآن حان وقت تركيب **PHP** .. أثناء كتابة هذه الكلمات **PHP 5.x** قد تم إصدارها وتعمل بدون مشاكل تذكر .. وجدير بالذكر انه يمكنك بالطبع استعمال إصدارات سابقة مثل **PHP 3.x** .

يمكنك تحميل البرنامج من موقع الشركة المنتجة :

http://www.php.net/downloads.php

قم باختيار النوع *Windows Binaries* package …

من أجل التعامل مع **PHP** تحتاج تركيب خادم ويب علي نظامك مثل :

Internet Information Services (IIS), Apache, Sambar

وسنشرح كيفية تركيب **IIS** و **Apache** في الصفحات التالية ..

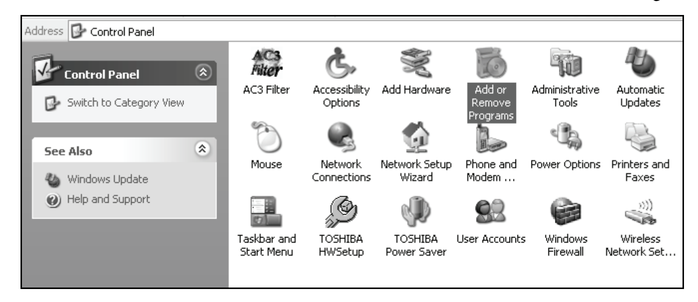
Installing IIS

تركيب خدمات معلومات الإنترنت

قم بالتوجه إلي **Control Panel** بالضغط علي قائمة **Start** ثم **Control Panel** :

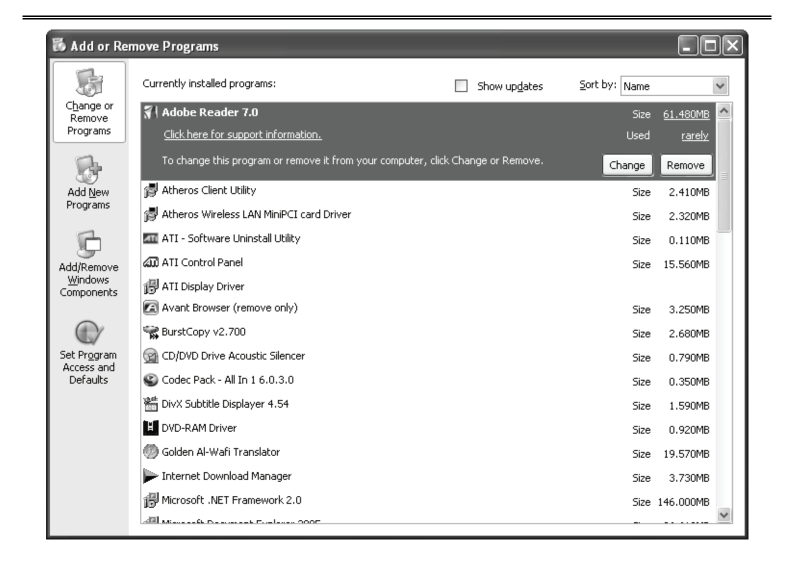


ستظهر لك النافذة التالية :



قم باختيار **Add or Remove Programs** ... بعد الضغط علي برنامج **Add or Remove Programs** ستظهر لك النافذة التالية ..

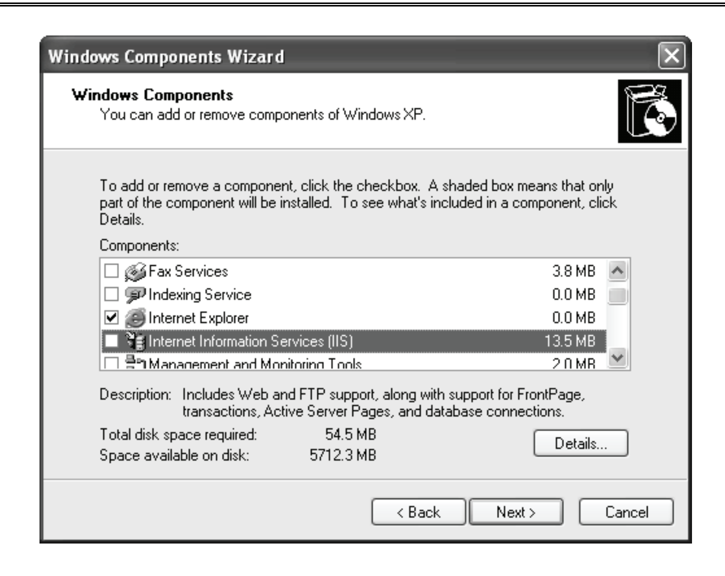
بالطبع أنت رأيت هذه النافذة كثيراً ...



لكننا لا نحتاج أن نقوم بإضافة برنامج أو إزالته .. نحن نريد إضافة مكون .. وذلك سيكون من خلال الضغط علي أيقونة برنامج **Add/remove Windows Components** ..



الآن سنقوم بتركيب المكون **IIS** ومشتقاته علي النظام ...



في الصورة السابقة أنت تجد أن المكون **IIS** غير منشط .. ولتنشيطه فقط

قم بالضغط علي صندوق الاختيار الخاص به ...

Internet Information Services (IIS)

قم بالضغط علي **Next** لبدأ عملية تركيب المكون ... سيقوم برنامج التركيب بنسخ الملفات الضرورية من اسطوانة الويندوز أو من علي القرص الصلب الخاص بك ....

هكذا نكون انتهينا من تركيب المكون **IIS** بنجاح ...



قم الآن بكتابة الرابط التالي في متصفحك :

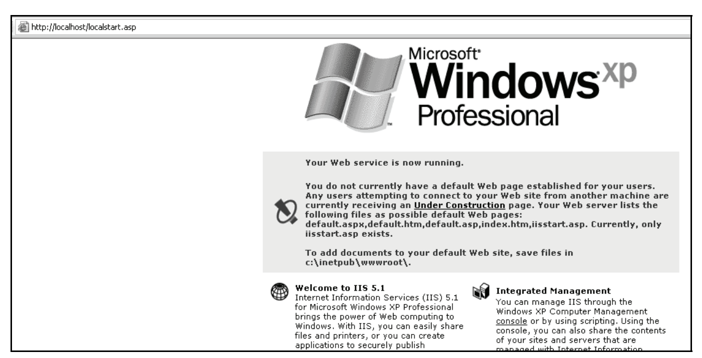
http://localhost/

ستجد أن المتصفح قد حولك إلي صفحتين .. الأولي :

http://localhost/localstart.asp

والثانية :

http://localhost/iishelp/iis/misc/default.asp

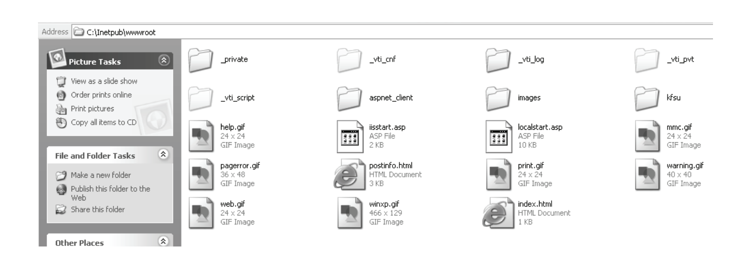


أين يتم وضع الصفحات التي نريد عرضها علي الخادم ؟

توجه إلي المسار التالي :

C:\Inetpub\wwwroot

ستجد مجموعة من الملفات التي تظهر في هذا المجلد ... منها الصفحات التي ظهرت لك في المتصفح ..



يمكنك الآن نسخ موقعك إلي هذا المجلد ثم تجربة استدعاؤه من المتصفح

وإن كنت تريد تجربة الخادم بشكل سريع يمكنك إنشاء ملف نصي جديد وكتابة نص بداخله :



ثم قم بتغيير اسمه إلي **Index.html** .. لاحظ إنه يجب عليك إزالة تفعيل خيار **Hide Extension** من **Folder Options** في **Control Panel** ..



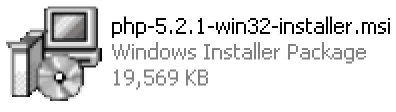
الآن قم باستدعاء الصفحة من المتصفح :



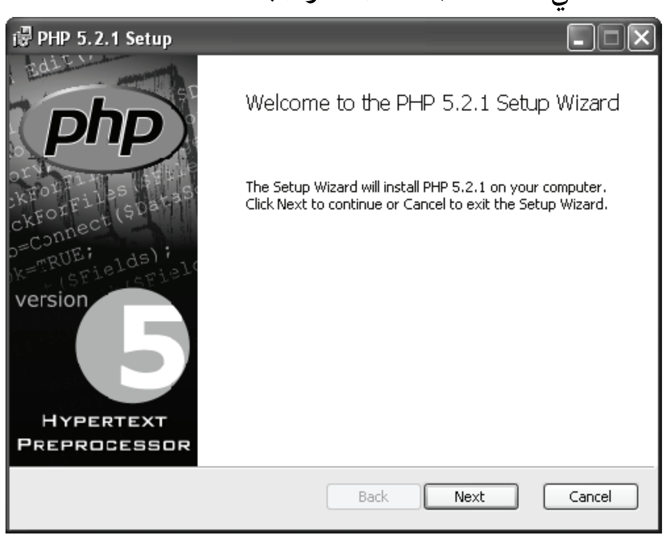
Installing PHP

تركيب البي أتش بي

نعود مرة أخري لتركيب برنامج **PHP** .. قم بتشغيل برنامج التركيب

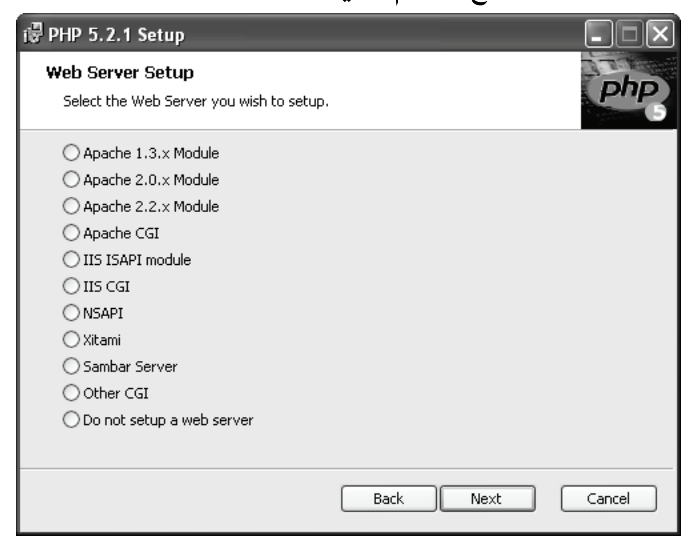


قم بالضغط علي **Next** لبدأ عملية التركيب :



قم بمتابعة التركيب الخطوات الأولي سـهلة حيـث تقـوم بتحديـد مكـان التركيب .. وما إلي ذلك .. الخطوات التي تلي ذلك تتسم بـبعض التقـدم لذا سأتناولها :

الخطوة التالية تحدد نوع الخادم الذي تعمل عليه ...



في حالة تركيبك لخادم أباتشي يمكنك اختيار نوع الخادم الذي تعمل عليه بمختلف إصداراته ...

ولأننا سنبدأ الشرح علي خادم **IIS** لذا سنختار **IIS ISAPI Module** كما تري في الصورة التالية :

IIS ISAPI module

أضغط **Next** لبدأ عملية التركيب ..

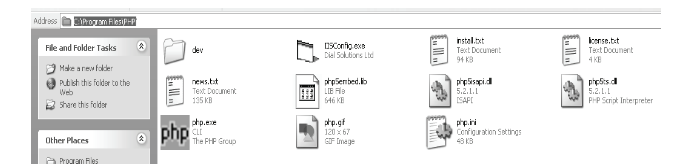
بالخطوة الأخيرة نكون قد نجحنا في تركيب برنامج **PHP** علي الخادم ..



إعداد **PHP** للعمل

قم بالتوجه إلي المكان الذي تم تركيب البرنامج إليه :

C:\Program Files\PHP



ثم قم بنسخ الملف **PHP5ts.dll** :



وقم بلصقه في المسار التالي :

C:\WINDOWS\system32

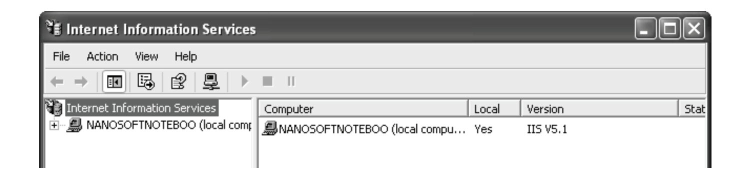
الآن توجه إلي لوحة التحكم **Control Panel** .. ثم اضغط علي **Administrative Tools**



ثم قم بتشغيل برنامج **Internet Information Services**



ستظهر لك النافذة التالية :

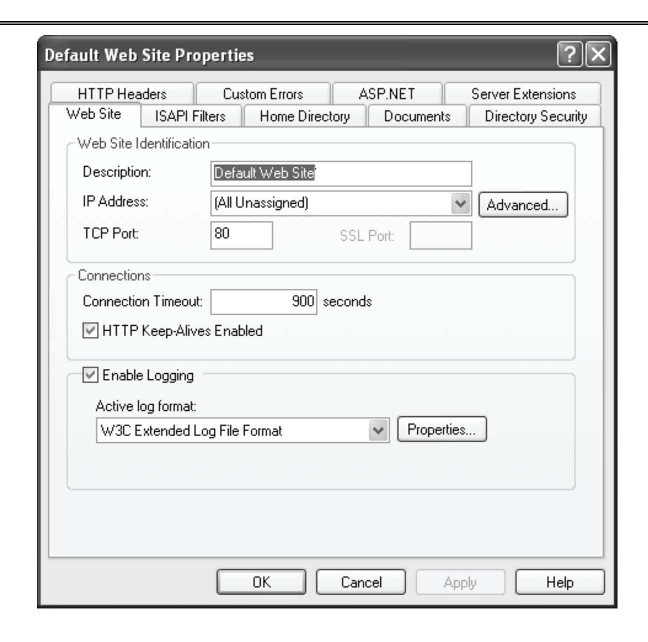


قم بالضغط علي أيقونة الجهاز في الغرض الشجري علي اليمين .. ثم اضغط علي **Web Sites** .. سيظهر لك الفرع **Default Web Site** كما تري في الصورة التالية :



قم بالضغط علي **Default Web Site** بالزر الأيمن للفارة .. ثم اختار من القائمة **Properties** ...

: ستظهر لك النافذة التالية

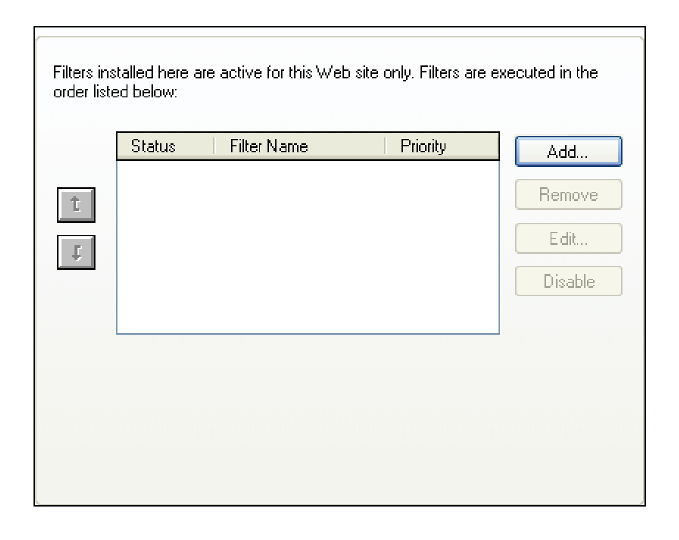


النافذة السابقة تسمح لك بإدارة مجموعة الخصائص للخـادم الخـاص بـك حيث يمكنك التحكم في اسم الدخول للخادم من خـلال **IP** تحـدده كما يمكنك تحديـد الـصفحة الرئيـسية لكافـة المواقـع الموجـودة علـي الخـادم الخاص به ...

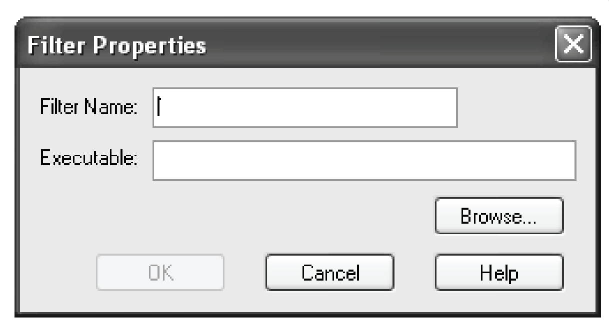
ما يهمنا الآن هو اللسان **Isapi Filters** .. قم بالضغط عليه ...

ISAPI Filters

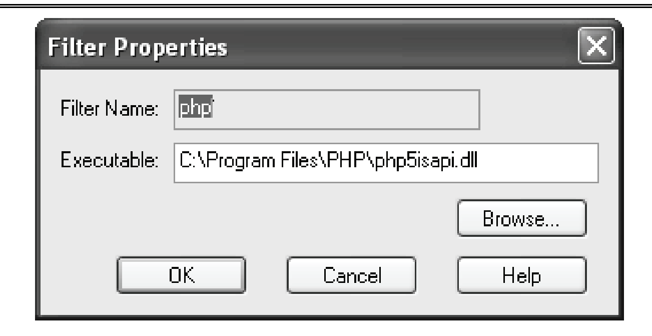
ستظهر لك مجموعة الخصائص التالية :



قم بالضغط علي الزر **Add** لإضافة فلتر جديد ... ستظهر لنا النافذة الصغيرة التالية :



قم بكتابة **PHP** في الحقل **Filter Name** .. ثم اضغط **Brows** اختار الملف **php5isapi.dll** من مكان تركيب الـ **PHP** ...



اضغط **OK** .. ستجد أن الفلتر قد تم إضافته بنجاح ..



بهذا الشكل نكن انتهينا من خطوة إضافة فلتر برنامج الـ **PHP** إلي الخادم ... وبقي لنا خطوتين أخرتين الأولي هي إضافة امتداد **PHP** ثم إضافة الصفحة الافتراضية **index.php** ...

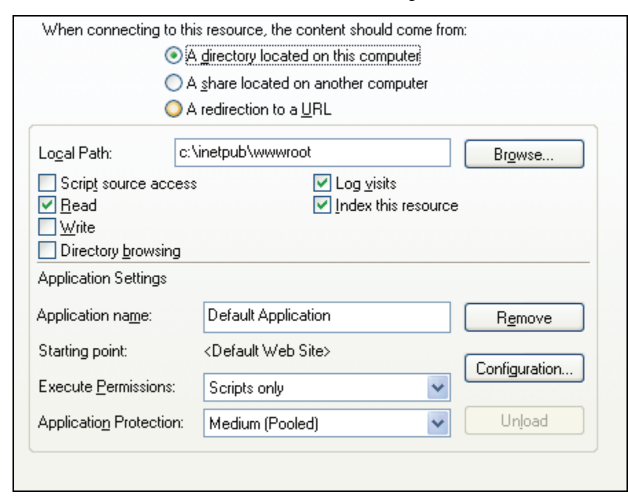
إضافة الامتداد ***PHP***

في خصائص الخادم قم بالتوجه إلي اللسان **Home Directory** ..



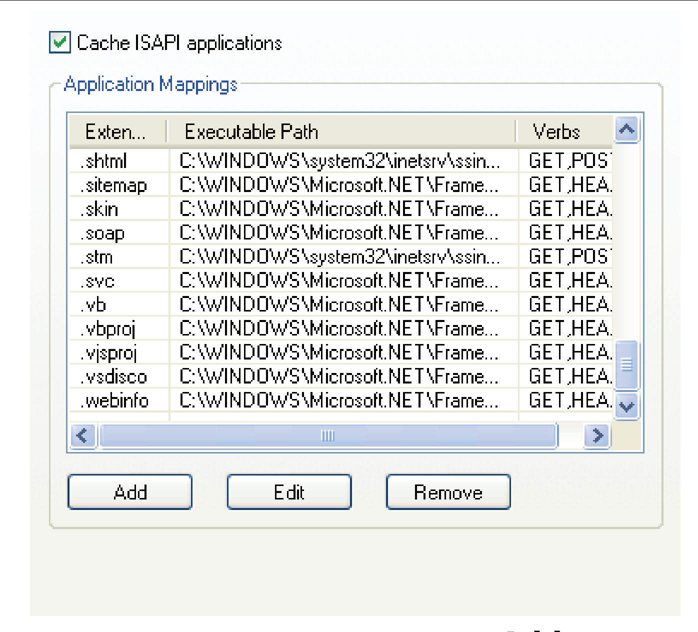
18

ستظهر لك الخصائص التالية :



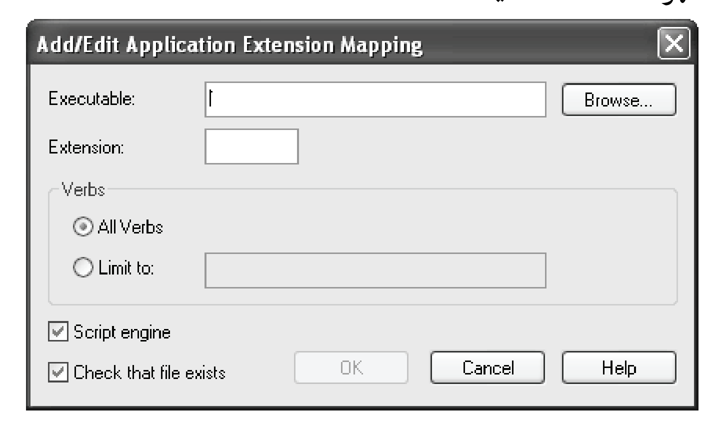
الآن سنتوجه إلي إعدادات التطبيق بالضغط علي الزر **Configuration** لنبدأ في إضافة امتداد الـ **PHP** ....

ستظهر لك النافذة التالية ...



قم بالضغط علي **Add** لإضافة امتداد جديد ....

ستظهر لك النافذة التالية ...



قم بالضغط علي **Browse** واختار ملف **php5isapi.dll** مرة أخري .. ثم اكتب **php.** في الحقل **extension** ...

Executable: "C:\Program File...\php5isapi.dll" Browse...
Extension: .php

اضغط **Ok** .. ستجد أن الامتداد قد تم إضافته لقائمة الامتداد الموجودة :

.php	C:\Program Files\PHP\php5isapi.dll	All

PHP إضافة صفحة افتراضية بامتداد

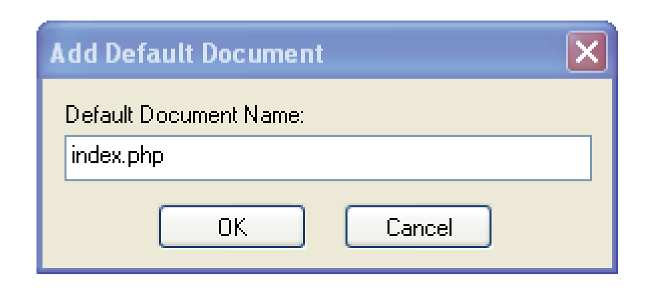
توجه إلي اللسان **Documents** ...

Documents

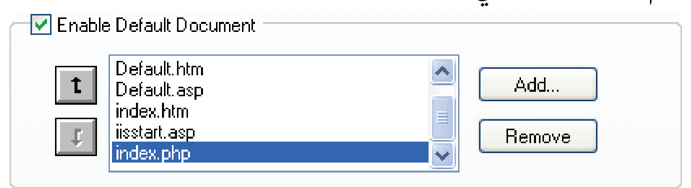
اضغط علي الزر **Add** في النافذة التالية ...

Enable Default Document
Default.aspx
Default.htm
Default.asp
index.htm
iisstart.asp
Add...
Remove
Enable Document Footer
Browse...

قم بكتابة **Index.php** في الحقل **Default Document Name** ..



الآن تم إضافة الملف إلي القائمة :

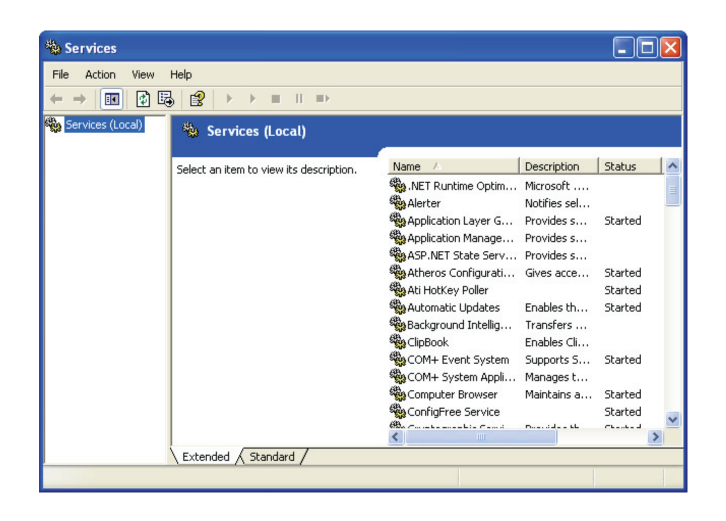


هكذا نكون انتهينا من ضبط إعدادات البرنامج ... بقي لنا أن نقوم بعمل إعادة تشغيل لخدمة الـ **IIS** حتى يستطيع أن يستوعب التغييرات التي قمنا بها ... ولعمل ذلك توجه إلي لوحة التحكم ثم **Administrative Tools** .. ثم اضغط علي البرنامج **Services** :

22



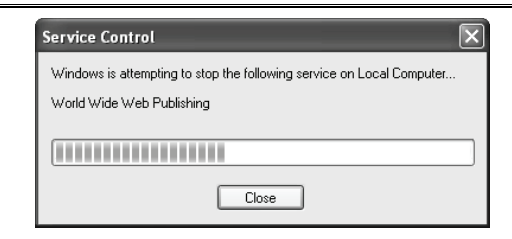
ستظهر لك النافذة التالية :



ثم توجه إلي نهاية القائمة لتجد الخدمة **World Wide Web Publishing** ..

World Wide Web Publishing Provides W... Started Automatic Local System

قم بالضغط علي الخدمة بالزر الأيمن للفارة ثم اختار **Restart** .. في حالة حدوث أي خطأ قم بعمل إعادة تشغيل للجهاز نفسه ...



سنتحدث هنا عن محررات لغة الـ **php** وهي كثيرة علي الإنترنت ومن أفضلهم :-

1- PHPEdit 1.0.4.82
2- php Expert Editor

البرنامج الأول :-

PHP Edit و بإصداره الجديد **PHPEdit 1.0.4.82**.. يعتبر هذا البرنامج من أقوى بيئات العمل **IDE** الخاصة بالـ **PHP** على **windows** يوفر هذا البرنامج ميزات عدة منها

1)) القائمة التلقائية للكلمات والتعليمات المحجوزة
2)) مصحح أخطاء **PHP** مدمج مع البرنامج.
3)) مدير للمهام **Task Manager**
4)) والكثير من الميزات الأخري.

للحصول على معلومات أكثر يمكنكم زيارة الموقع الأصلي للبرنامج

/http://www.waterproof.fr

لتحميل البرنامج اذهب إلى الوصلة وثم اختر السير فر الذي تريد التحميل منه للتحميل :-

http://www.waterproof.fr/products/PHPEdit/download.php

صورة للبرنامج



البرنامج الثاني هو php Expert Editor ..

وهو أيضاً من أقوي البرامج المستخدمة وهو الذي سوف نقوم باستخدامه في عملنا وستجد شرحه وشرح كيفية الحصول عليه وتثبيته والعمل عليه في فصل تثبيت أول منتدى لك **vb229** .

عبارات في اللغة ***php***

– هناك كلمات في لغة الـ **PHP** تستخدم لأعمال معينة ، فمثلاً هناك العبارة **IF** التي تحدد شرط معين لتفعيل الأمر والعبارة **Else** التي تتيح إجراء ثان و عبارات أخري سنتعرف عليها في هذا الفصل .

IF

إن الـ **PHP** تتبع نفس الأسلوب في كتابة الـ **if** مثل الفيجوال بيسيك والديلفي وغيرها من لغات البرمجة، فيمكن تحديد شرط معين مقترن بالـ **if** وبالتالي إذا كان الشرط صحيحاً (**true**) فسيتم تنفيذ الأسطر المحددة ، وبتفصيل أكثر يجب وضع الشرط بين قوسين ( ) ، ووضع الأسطر المطلوب تنفيذها بين العلامـات { } ، مع ملاحظة أنه يمكن التخلي عن العلامات { } في حال وجود كتابة الأكواد في سطر واحد .

مثال علي كود عبارة الـ **IF** في الفيجوال بيسيك :

CODE

```
If Text1.Text = "ahmed fawzy" Then Msgbox "password
correct"
Else
Msgbox "password Wrong"
End If
```

شرح الكود :

– هنا يقوم البرنامج بمعرفة إذا كان تم كتابة **Ahmed fawzy** فلو حدث سيتم إظهار رسالة تأكيد وإذا كانت غير ذلك فيتم إظهار رسالة خطأ الباسوورد .

– فلنفترض وجود نموذج اشتراك أو نموذج بريد ، يحتوي على الاسم والبريد والرسالة ومعلومات أخري ، ونرغب في معرفة ما إذا كان المرسل قد ملأ جميع الحقول وبالتالي إرسال الرسالة ، أو انه لم بفعل ذلك وهناك حقول ناقصة و بالتالي عرض رسالة (فضلاً قم بتعبئة البيانات التي لم تكتمل) ، لعمل ذلك نحتاج معرفة أسماء المتغيرات في النموذج ، ولذلك فلنفترض أن المتغيرات كالتالي :

(الاسم **$name**) ، (البريد **$email**) ، (الرسالة **$later**)

، ولعمل الشرط الأول (إذا كان الاسم لم يُدخل فلن يتم إرسال الرسالة) :

CODE

```
<?
if ( $name == "" )
echo "فضلاً قم بتعبئة البيانات الناقصة ";
?>
```

والمعنى أنه إذا كان المتغير **name$** لا يحتوي على أي قيمة (أي فراغ) فسيتم تنفيذ السطر التالي وطباعة الجملة ، مع ملاحظة أن المطلوب تنفيذه هو سطر واحد فقط ولذلك لم نستخدم { } ، بل في حالة وجود أكثر من سطر يجب استخدامها كالتالي

CODE

```
<?
if ( $name == "" )  {
echo "فضلاً قم بتعبئة البيـانات الناقصة<br>";
echo "لم تقم بإدخـال الاسـم الخاص بك";
}
?>
```

ELSE

هذه العبـارة تتيح إمكانية وجود أجراء ثاني لعدم تحقق الشرط ، ففي مثالنا السابق كان الأجراء طباعة الجملة إذا تحقق الشرط ، ولكن في حالة عدم تحقق الشرط فلن يكون هناك أجراء لتنفيذه ، بل إن الأجراء سيتم تنفيذه إذا تحقق الشرط ومن ثم سيتم أكمال بقية الأسطر ، ونجد هذا المثال مثل المثال السابق ذكره في لغة الفيجوال بيسيك، وفي حالة مثل هذه الحالة يتم استخدام الـ **else** لوضع أجراء آخر في حالة عدم تحقق الشرط ، المثال :

CODE

```
<?
if ( $name == "" )  {
echo "فضلاً قم بتعبئة البيـانات الناقصة";
}
else
{
echo "تم إرسـال الرســالة للادمين ، شـكرا لك";
}
?>
```

في هذا المثال سيتم إظهار الرسالة(فضلاً قم بتعبئة البيـانات الناقصة) إذا تحقق الشرط أن المتغير **name$** لا يحتوي على أي قيمة ، وسيتم طباعة الجملة (تم إرسال الرسـالة ، شكرا لك) في حالة عدم تحقق الشرط ، أي في حالة وجود قيمة في المتغير **name$** ، مع ملاحظة أن هذا المثال يحتوي على شرطين وليس شرط واحد ، فالظاهر هو شرط واحد ("" == **name$**) ولكن العبـارة **else** تعتبر شرطاً بحد ذاتها ولو لم يكن هذا الشرط مكتوباً ، وكما هو واضح فمعنى هذا الشرط هو (إذا كان غير ذلك) فقم بإظهار الرسالة .

ELSEIF

في العبارة السابقة ذكرنا أنه يوجد شرطين و إجراءين ، أحد هذين الشرطين غير مكتوب بل هو مفهوم من أدراج العبـارة **else** ، وفي

حالات كثيرة لا يكفينا مجرد شرطين و إجراءين لإتمام بعض السكربات المعقدة ، فلذلك يمكن نستخدم العبـارة **elseif** مع الـ **if** لعمل مثل هذه السكربات .

بافتراض أن المتغير (**$Age**) هو عداد الزوار ، فالمثال التالي يبين ما تم شرحه سابقاً :

CODE

```
؟>
if ( $ Age == 13 )  {
echo " السـن : ثلاثة عشر عام "؛
{
elseif ( $ Age == 14 )  {
echo " السـن : أربعة عشر عام "؛
{
elseif ( $ Age >== 15  )  {
echo "السـن : $ Age السـن"؛
{
else {
echo "عدد الزوار : $ Age السـن"؛
{
<؟
```

كما هو واضح في المثـال السابق سيتم ما يلي :

الشرط : السن يساوي 13 عاماً

الإجراء : إظهار رسالة (السن : 15 عاماً)

الشرط : السن يساوي 14

الإجراء : طباعة (السن : 15 عاماً)

الشرط : السن يساوي 15

الإجراء : طباعة (السن : 15 عاماً)

الشرط : السن لا يحقق أي من الشروط

الإجراء : طباعة (السن : (السن) عاماً)

WHILE

وهي ابسط حلقات التكرار ، بحيث تأخذ شرط واحد فقط وتبني على تنفيذ ما بين علامات الشروط { } ، وتختلف عن عبارة الـ **if** هو أنها ستقوم بتنفيذ الإجراءات طالما كان الشرط صحيحاً ، وهذا يعني احتمال تنفيذ الإجراء أكثر من مرة ، وهذه الدالة مفيدة في أدراج الحقول من الجداول وغيرها من الاستخدامات ، بحيث لو افترضنا وجود جدول معين في قاعد بيانات ونريد أدراجه في صفحة **PHP** ، فسيكون من أهم خيارات الاستخدام هذه الدالة – قم بقراءة فصل التعامل مع قواعد البيانات – و في الوقت الحالي سأذكر مثال بسيط على هذه الدالة لفهم طريقة استخدامها :

CODE

```
<?
$total = 20;
while ( $total <=40 )
{
echo "العدد أقل من 40<br>";
$total +=20;
}
?>
```

FOR

استـخدام حلقـة التكـرار **while** كانـت القيمـة الابتدائيـة للمـتغير (**$total**) في سطر ، والشرط في سطر والزيـادة على المتغير في سطر أخـر ، وبالتالي زيادة في عدد الأسطر عن ما يمكن استخدامه مع حلقة التكـرار **for** ، فالمثـال التالي يبين طريقة أخـرى لاسـتخدام مثـال الــ **while** بطريقة أسهل :

CODE

```
<?
for ( $total = 20;  $total <=40; $total +=20 )
{
echo "العدد أقل من 40<br>";
}
?>
```

وللتوضيح فان تركيب الـ **for** هو على الشكل التالي :

CODE

```
for ( القيمة الافتراضية؛ الشـرط؛ مقدار الزيـادة ;)
{
الإجراء المطلوب تنفيذه
}
```

DO WHILE

وهي تشبه في عملها عمل عبارة الـ **while** وتختلف عنا بأنها تتأكد من الشرط وصحته من عدمها يتم بعد تنفيذ الأجـراء ولـيس قبلـه كمـا في الــ **while** وكمثال عليها :

CODE

```
<?
$total = 20;
do
{
echo "40 العدد أقل من<br>";
$total +=20;
}
while ( $total <= 40 );
?>
```

Dealing With Files

التعامل مع الملفات

لغة الـ **php** توفر طرقتين للتعامل مع الملفات :

الطريقة الأولي : تستخدم **file handle** .

الطريقة الثانية : تستخدم مسار الملف **path** .

ملحوظة

File Handle هو عدد صحيح يقوم بتعريف الملف المراد فتحه حتى يتم إغلاقه .

فتح وإغلاق الملفات

قبل كل شيء :

قم بإنشاء ملف مساره **d:\file.txt** واكتب فيه أكثر من سطرين .

Fopen

قم بكتابة الكود التالي في صفحة ذات امتداد **php** وقم بحفظها وقم بتشغيل سيرفر أباتشي ثم قم بتشغيل الصفحة التي قمت بإنشائها منذ قليل .

مثال (1) :

CODE

```
$fp=fopen ("d:\file.txt", "r");
if (!$fp) die ("لم نتمكن من القراءة من الملف");
```

مثال (2) :

CODE

```
If  (!($fp=fopen ("d:\file.txt", "r"))) die (" لم نتمكن من القراءة
من الملف");
```

- المتغير $fp :

هي رقم المقبض التي يخزن فيها المعلومات الخاصة بالملف من مساره أو إمكانية فتحه أم لا وهكذا .

Fclose

عند الانتهاء من التعامل مع الملف يجب إغلاقه لكي يتم حفظ التعديلات التي أجريت عليه ، تقوم الدالة **fclose**() بإغلاق الملف عندما تريد إغلاقه وهي تحتاج إلي معامل واحد فقط وهو مقبض الملف الذي تريد إغلاقه

مثال:

CODE

```
Fclose ($fp) ;
```

قراءه وكتابه الملفات

Fread :

تقوم هذه الدالة بقراءة البيانات الموجودة في الملف ووضعها بمتغير .

مثال :-

CODE

```
$fp=fopen("data.txt","r");
$data=fread($fp,10);
```

Fwrite :

تقوم هذه الدالة بالكتابة إلى الملف وتحتاج إلى عاملين وهي مقبض الملف

والقيمة المراد كتابتها إلى الملف :

CODE

```
Fwrite ($fp, "PHP");
```

وهناك معامل ثالث لهذه الدالة يحدد كم حرفا سنقوم بكتابته من القيمة الحرفية الموجودة في المعامل الثاني فلو مثلا كتبنا فسوف يتم كتابه أول حرف فقط .

CODE

```
Fwrite ($fp, "PHP",1);
```

وهناك دالة مشابهة للدالة السابقة وهي **Fputs** وهي تقوم بنفس الوظائف ولها نفس الخصائص والمعاملات

قراءة وكتابة الحروف في الملفات

قبل كل شيء :-

قم بإنشاء ملف مساره **d:\file.txt** واكتب فيه أكثر من سطرين .

Feof

تقوم هذه الدالة بمعرفة إذا ما كنا قد وصلنا لنهاية الملف عند قراءته وتقوم بإرجاع (**true**) عند الوصول إلى نهاية الملف أو حدوث خطأ في قراءة ملف

مثال :

CODE

```
<?
$fp= fopen("d:\file.txt","r");
While (!feof($fp))
{
 $char=fgetc($fp);
     echo $char;
} ?>
```

Fgets

– من أجل فهم هذه الدالة قم بوضعها تحت التنفيذ .

مثال :

CODE

```
$fd = fopen ("/tmp/inputd:\file.txt", "r");
while (!feof ($fd)) {
    $buffer = fgets($fd, 4096);
    echo $buffer;
}
fclose ($fd);
```

القراءة داخل الملفات

File

وعملها هو قراءة ما بداخل الملف وتخزينه في مصفوفة ، تحتاج فقط إلي مسار الملف **path** .

مثال :

CODE

```
<?
$fcontents = file ('d:\file.txt');
while (list ($line_num, $line) = each ($fcontents)) {
    echo "<b>Line $line_num:</b>  $line  <br>\n";
}
?>
```

Readfile

وظيفتها هي قراءة جميع محتويات الملف وتطلب مسار الملف فقط وتقوم بقراءة كامل محتويات الملف ثم طباعتها .

CODE

```
<?
Readfile ("d:\file.txt");
?>
```

جلب معلومات الملف

File_exists

وظيفتها التأكد من وجود الملف في المسار المحدد أم لا وهي تطلب مسار الملف فقط ، وتقوم بإرجاع **true (1**) إذا كان الملف موجوداً و **false** إذا كان الملف غير موجود وعند تشغيل الصفحة بإست دعاؤها من الأباتشي ستجد الصفحة فارغة:

CODE

```
<?
$Th=File_exists("d:\file.txt");
echo $Th ;
?>
```

Filesize

تقوم هذه الدالة بالحصول علي حجم الملف .

إجراء عمليات الحذف والتعديل للأسماء

Copy, Rename & Delete Operations

في لغة الـ **php** يمكننا إجراء عمليات علي الملفات مثل الحذف وإعادة التسمية وغيرها من العمليات التي قد نحتاجها أثناء عملنا في تصميم صفحة .

Copy ()

تطلب شيئين الأول هو مسار الملف الرئيسي والأخر مسار الملف الذي سيتم إنشاءه من خلال الدالة المستخدمة

CODE

```
<?
if (!copy($file, $file.'.bak')) {
    print ("failed to copy $file...<br>\n");
}
?>
```

Rename

نستطيع الآن استخدام هذه الدالة لإعادة تسمية الملف وهي تحتاج إلي قيمتين حرفتين وهي المصدر الملف أو مكانه واسمه الرئيسي ثم الاسم الجديد الذي تريد إعادة التسمية به ..

مثال :

CODE

```
<?
Rename ('d:\file.txt','newd:\file.txt');
?>
```

Unlink()

وظيفتها حذف الملف الذي حدد من خلال المسار المعطي من قبل .

CODE

```
<?
unlink ('d:\file.txt');
?>
```

التعامل مع المجلدات

Dealing With Folders

هناك أيضاً دوال للتعامل مع المجلدات مثل : **Opendir** و **Closedir** و **Readdir** وغيرهم الذين سنتناولهم بالشرح .

Opendir

وظيفة هذه الدالة فتح المجلد .

Closedir()

وظيفة هذه الدالة إغلاق المجلد المفتوح وتحتاج فقط إلى مقبض المجلد .

Readdir

وظيفة هذه الدالة قراءة المدخل الحالي للمجلد .

Rewindir

وظيفة هذه الدالة إرجاع المدخل من الصفر .

Chdir

وظيفة هذه الدالة الانتقال إلي مجلد آخر ، وتتطلب المسار للمجلد الذي تريد الانتقال إليه .

Rmdir

وظيفة هذه الدالة أن تقوم بمسح مجلد ، ولكن يجب أن يكون المجلد خاليا من أي ملفات أو مجلدات ، وتتطلب مسار المجلد الذي تريد مسحه .

Mkdir

وظيفة هذه الدالة إنشاء مجلد جديد وتتطلب أن يكون هذا المجلد غير موجود مسبقاً وتحتاج إلي قيمتين وهما اسم المجلد الجديد مع مساره ، والترخيص المطلوب له .

Dirname

وظيفة هذه الدالة إعطائنا اسم المجلد الحالي الذي فيه الملف ، وتحتاج إلي مسار الملف .

مثال :-

أنـشئ مجلـد اسمـه **files** في مجلـد الــ**htdocs** بمجلـد الأباتـشي **Apache** وقـم بإنـشاء عـدة ملفـات داخلـه ، ثـم أنـشئ ملـف اسمـه **play.php** أو أي أسم أخر في مجلد الــ**htdocs** واكتـب بـه الكـود التالي ثم أستدعيه بعد تشغيل الأباتشي طبعاً :

CODE

```
<?php
if ($dir = @opendir("/files")) {
  while($file = readdir($dir)) {
    echo "$file\n";
  }
  closedir($dir);
}

?>
```

أول كود PHP

برجاء مراجعة ملحق الكتاب علي موقع دار النشر وأقرء الجزء الخاص بتركيب PHP

الآن سنتكلم عن أول كود بي اتش بي لك .. لا تقلق ستكون ذا خبرة جيدة في هذه اللغة بعد انتهاؤك من صفحات هذا الكتاب ...

كما سيكون من الظلم بعد كل هذه البرامج الذي ركبتها ألا تري نتيجة لعملك ولو نتيجة بسيطة ..

قم بفتح محرر نصوص أو محرر ملفات **PHP** .. سنتكلم عن برنامج متخصص لتحرير هذه الملفات عندما نتكلم عن **PHP** بتخصص ..
أول كود لك سيكون تحديد الوقت علي الخادم :

Code :

```
<html>
<head>
<title>Today's Date</title>
</head>
<body>
<p>Today's Date (according to this Web server) is
<?php
echo( date('l, F dS Y.') );
?></p>
</body>
</html>
```

قم بإنشاء ملف باسم **today.php** في المسار :

C:\Inetpub\wwwroot

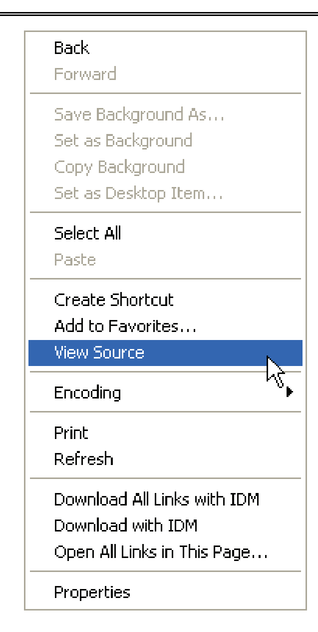


واكتب الكود السابق ... الآن توجه إلي المتصفح .. ثم اكتب رابط الصفحة التي ناشاتها ... أرجو أن تكون النتيجة كالتالي :

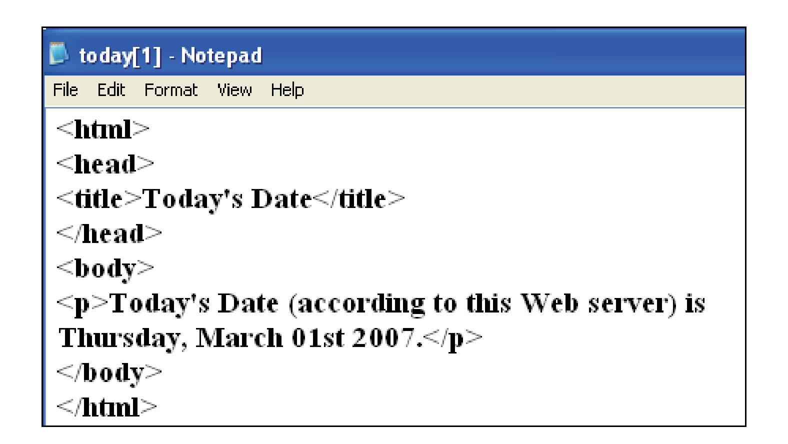


Today's Date (according to this Web server) is Thursday, March 01st 2007.

الآن قم بالضغط علي الصفحة بالزر الأيمن للفارة لتظهر لك القائمة التالية فتختار منها **View Source** ...



سيظهر لك محرر النصوص **Notepad** ...




```
<html>
<head>
<title>Today's Date</title>
</head>
<body>
<p>Today's Date (according to this Web server) is
Thursday, March 01st 2007.</p>
</body>
</html>
```

ستجد أن الكود المستخدم مختلف عن الكود الذي كتبناه ....

Code :

```
<html>
<head>
<title>Today's Date</title>
</head>
<body>
<p>Today's Date (according to this Web
server) is
Thursday, March 01st 2007.</p>
</body>
</html>
```

ستجد أن الكود الذي كتبناه يعمل علي جانب الخادم بحيث يأتي بقيمة يكتبها بداخل الكود ..

قواعد البيانات في PHP ..

ملحوظة :

سنذكر في فصل لاحق كيفية تركيب وإنشاء والتعامل مع قواعد بيانات **MySql** ..

دوال الاتصال بقواعد البيانات

تعريف الدالة :-

الدوال (Function)

تقوم بوظيفة معينة ، كما توجد إمكانية إنشاء دوال تؤدي وظيفة خاصة

وحديثنا هنا عن هذا النوع من الدوال (كيفية إنشاء دوال)

الدالة تقوم بتنفيذ شيء معين حيث تأخذ (متغيرات – معطيات) ثم

تقوم بمعالجة هذه المتغيرات وتخرج قيمة أخرى .

الشكل العام للدالة :-

CODE

```
Function( اسم الدالة ( المعطيات – المتغيرات – البارامتر
{
هنا يتم كتابة الكود
Return ( المعطيات – المتغيرات – البارامتر ) ;
}
```

تعريف الدالة :-

لكي نقوم بتعريف دالة نكتب كلمة **function** بعدها اسم الدالة وبعد

الاسم نكتب المعطيات – المتغيرات بين قوسين .

مثال :-

CODE

```
<?
(Function dalla(Integer
?>
```

حيث **dalla** هو اسم الدالة ، وبالتأكيد يمكن أن يكون أي اسم .

(**Integer**) هو (المتغير – المعطى – البارامتر) ، أي اسم من هذه

كما تحب أن تسميه .

– بعد ذلك نقوم بكتابة كود الدالة (عمل الدالة) بين العلامتين { } ،

كما يجب

أن ننهي الدالة بكلمة **return** لإعلام الدالة بأن وظيفتها قد انتهت

بالإضافة إلى

ذكر اسم المتغير المذكور في تعريف الدالة سابقا .

مثال :

CODE

```
<?
;(Return(Integer
?>
```

أظهر نتيجة الدالة :-

نستخدم الأمر الخاصة بالطباعة **echo** أو **print** وبعده طبعا اسم الدالة .

مثال :

CODE

```
<?
echo dalla(5);
print dalla(5);
?>
```

دوال قواعد البيانات

هنا سنستعرض الدوال والأوامر التي تمكننا من الاتصال مع قواعد البيانات **MYSQL** .

MySQL_connect :-

الدالة :-

CODE

```
integer mysql_connect(host, username, password);
```

تقوم هذه الدالة بالاتصال مع قاعدة البيانات وتعيد لك رقم يفيدك إذا كان لديك أكثر من اتصال بقواعد البيانات

وهي تحتاج إلي ثلاث متغيرات :-

اسم السير فر الذي سيتم الاتصال به.

اسم مستخدم قاعدة البيانات.

كلمة المرور الخاصة به.

مثال :

CODE

```
<?
$link = mysql_connect("localhost","kito","mypages");
?>
```

* **كيف يمكن التأكد من نجاح الدالة ؟**

– يمكنك التأكد من نجاح الدالة من خلال جعل البرنامج الذي تقوم بتصميمه إلي طبع كلمة النجاح في الاتصال عند الاتصال وكلمة فشل في الاتصال عند الفشل في الاتصال ويمكنك هذا من خلال الكود التالي :–

CODE

```
<?
$link = mysql_connect("localhost","kito","mypages");
or die ("فشل الاتصال")
Print ("نجاح الاتصال")
?>
```

MySQL_Pconnect :-

الدالة :-

CODE

```
integer mysql_pconnect(host, username, password);
```

هذه الدالة تقوم بما تقوم به الدالة السابقة إلا أنه لا يتوجب عليك إغلاق الاتصال

مثال:

CODE

```
<?
$link = mysql_pconnect("localhost","kito","Pass");
?>
```

-: MySQL_Select_DB

الدالة :-

CODE

```
boolean mysql_select_db(string database, integer link);
```

تقوم هذه الدالة باختيار قاعد البيانات المحدد لها. مثال:

CODE

```
<?
mysql_select_db(string database, integer link);
?>
```

-: MySql_Query

الدالة :-

CODE

```
integer = mysql_query(string query, integer link);
```

تقوم هذه الدالة بما تقوم به الدالة **mysql_db_query** تقريباً إلا أن الدالة **mysql_query** يقتصر عملها على قاعدة البيانات المحددة بالدالة **mysql_select_db** .

في حالة عدم تمرير رقم الاتصال فستعمل الدالة على الاتصال الأخير.

مثال:

CODE

```
<?
$link = mysql_connect("localhost","kito","Pass");
$query = "DELETE FROM magazine";
$result = mysql_query($query, $link);
?>
```

MySQL_DB_Query :-

الدالة :-

CODE

```
boolean mysql_db_query(string database, string query,
integer link);
```

تقوم هذه الدالة بتنفيذ سطر **SQL** على قاعدة البيانات المفتوحة بالمعطى **database**

مثال:

CODE

```
<?
$link = mysql_connect("localhost","kito","Pass");
$Query = "DELETE FROM magazine";
$result = mysql_db_query("kito", $Query, $link);
?>
```

-: MySQL_Error

الدالة :-

CODE

```
string mysql_error(integer link);
```

تعيد هذه الدالة رسالة الخطأ الحاصل في قاعدة البيانات .

-: MySQL_ErrorNo

الدالة :-

CODE

```
integer mysql_errno(integer link);
```

تقوم هذه الدالة بإعادة رقم آخر خطأ حدث في التعامل مع قاعدة البيانات.

My sql_creat_db
الدالة :-

CODE

```
boolean mysql_create_db(string databasename, integer link);
```

تقوم هذه الدالة بإنشاء قاعدة بيانات جديدة مرر لها اسم قاعدة البيانات ورقم الاتصال العائد من الدالة **mysql_connect** أومن الدالة **mysql_pconnect** ..

مثال:

CODE

```
<?
// الإتصال بقاعدة بيانات az حيث أن الفراغ هو الباسورد اسمها
$link = mysql_pconnect("localhost", "kito", "mypages")
;
// انشاء قاعدة بيانات جديدة
if (! mysql_create_db($link, "kito"))
{
    print("عملية فاشلة")
    exit();
}
?>
```

Mysql_drop_db

الدالة :-

CODE
```
boolean mysql_drop_db(string databasename, integer link);
```

تقوم هذه الدالة بحذف قاعدة البيانات المحددة بالمعطى

databasename ..

Mysql_list_dbs

الدالة :-

CODE
```
integer mysql_list_dbs(integer link);
```

تقوم هذه الدالة بإعادة مؤشر لكل قواعد البيانات الموجودة في الخادم لغرض استعمالها مع الدالة **mysql_fetch_row** وأمثالها .

Mysql_field_seek

الدالة :-

CODE
```
boolean mysql_field_seek(integer result, integer field);
```

تقوم هذه الدالة بتحديد الحقل المرر إليها رقمه .

مثال :

CODE

```
<?
// الإتصال بقاعدة vb229 حيث أن الفراغ هو الباسورد
بيانات اسمها
$dbLink = mysql_pconnect("localhost","vb229","");
//  Authors أختيار قاعدة البيانات
mysql_select_db("Authers", $dbLink);
// Adress اختيار جميع الحقول من الجدول
$Query = "SELECT * FROM adress";
$result = mysql_query($Query, $dbLink);
//  الانتقال إلى الحقل الثاني اعتماداً على عملية
الاختيار
mysql_field_seek($reslut, 1);
?>
```

Mysql_field_name

الدالة :-

CODE

```
string mysql_field_name(integer result, integer feild);
```

تعيد هذه الدالة اسم الحقل المحدد بالرقم المرر إليها والذي يبدأ بالرقم صفر للحقل (العمود) الأول . مثالها سيأتي بعد قليل .

Mysql_field_len

الدالة :-

CODE

```
string mysql_field_type(integer result, integer feild);
```

تعيد هذه الدالة نوع الحقل المحدد بالرقم الممرر إليها والذي يبدأ بالرقم صفر للحقل (العمود) الأول . المثال سيأتي بعد قليل أيضاً ..

Mysql_field_len

الدالة :-

CODE

```
string mysql_field_len(integer result, integer feild);
```

تعيد هذه الدالة طول الحقل بالبايت المحدد بالرقم المرر إليها والذي يبدأ بالرقم صفر للحقل (العمود) الأول .

Mysql_field_flags
الدالة :-

CODE

```
string mysql_field_flags(integer result, integer feild);
```

تعيد هذه الدالة وصف الحقل المحدد بالرقم المرر إليها والذي يبدأ بالرقم صفر للحقل (العمود) الأول .

Mysql_list

الدالة :-

CODE

```
mysql_list(string database, string table, integer link);
```

مثال شامل:

CODE

```
<?
// الإتصال بقاعدة vb229 حيث أن الفراغ هو الباسورد بيانات اسمها
$link = mysql_pconnect("localhost", "vb229", "");
// ترتيب الحقول وجلبها
$result = mysql_list_field("mag", "table", integer lin
k);
// حلقة تكرار للمرور على كل حقل
for (Integer = 0; Integer < mysql_field_num($result);
Integer++)
{
```

```
    print(mysql_field_name($result, $i);
    print(mysql_field_type($result, $i));
    print(mysql_field_len($result, $i));
    print(mysql_field_flags($result, i));
}
?>
```

Mysql_fetch_field

الدالة :-

CODE

```
<?
object mysql_fetch_field(integer result, integer field);
?>
```

تستخدم هذه الدالة لتحصل على معلومات حول حقول الجدول المراد،

الحقول ترقم بداية من صفر .

Mysql_fetch_lenghts

الدالة :-

CODE

```
<?
array mysql_fetch_lengths(integer result);
?>
```

استخدم هذه الدالة لتعيد مصفوفة تحتوي على الطول الأقصى لكل حقل

محدد في المعطي**result**.

CODE

```
<?
```

```
//Connect to server as kito no password
$link = mysql_pconnect("localhost","kito","");
//Select th magazine database
mysql_select_db("magazine",$link);
//Get name and nomber from magazine
$Query = 'SELECT name, nomber FROM magazine';
$result = mysql_query($Query, $link);
$length = mysql_fetch($result);
//Print length of the third columnh
print($lengths[2]);
?>
```

MySQL_Fetch_Array :-

الدالة :-

CODE

```
<?
array mysql_fetch_array(integer result);
?>
```

هذه الدالة تعيد مصفوفة تحتوي على قيم سجل وتنقل المؤشر إلى السجل التالي. مثال:

CODE

```
<?
//Connect to server as kito no password
$link = mysql_pconnect("localhost","kito","");
//Select th magazine database
mysql_select_db("magazine",$link);
//Get name and nomber from magazine
$Query = 'SELECT name, nomber FROM magazine';
$result = mysql_query($Query, $link);
//Get every row
while($row=mysql_fetch_array($result, MYSQL_ASSOC)){
 //Print mane and nomber
   print({$row["nomber"]}={$row["name"]});
```

```
}
?>
```

-: MySQL_Fetch_Object

الدالة :-

CODE

```
<?
object mysql_fetch_object(integer result)
?>
```

هذه الدالة تشبه الدالة **mysql_fetch_array** إلا أنها تعيد كائن.

عند استدعاء الدالة ينتقل المؤشر إلى السجل التالي في الجدول، وإذا ةصل إلى نهاية الجدول ثم استدعيت الدالة مرة أخرى فإنها تعيد القيمة **FALSE** مثال:

CODE

```
<?
while($row=mysql_fetch_object(result)){
 //print nomber and name
   print ("$row->nomber, $row->name")
}
?>
```

-: MySQL_Fetch_Row

الدالة :-

هذه الدالة تعيد مصفوفة تحتوي على قيم حقول سجل من الجدول وكل استدعاء يعيد قيمة الحقول في السجل التالي في الواقع هذه الدالة تشبه الدالتين السابقتين. مثال:

CODE

```
<?
while($row=mysql_fetch_row(result)){
 //print nomber and name
   print ("$row[0], $row[1]")
}
?>
```

MySQL_Change_User :-

الدالة :-

CODE

```
<?
mysql_change_user(string user, string password, string db, inte
ger link);
?>
```

استخدم هذه الدالة لتغيير مستخدم قاعدة بيانات المتصل بها.

المعطيان **db, link** اختيارية وفي حالة فقدهما يستعاض عنهما بالاتصال الحالي.

هذه الدالة تتطلب إصدار **MySQL 3.23.3** أو ما بعدها.

الدالة MySQL_Close :-

CODE

```
boolean mysql_close(integer link);
```

تقوم هذه الدالة بقطع (إغلاق) قاعدة البيانات ، مرر لها رقم الاتصال المعاد من الدالة **mysql_connect**

مثال:

CODE

```
<?
// الاتصال بقاعدة البيانات ..
$link = mysql_connect("localhost","mag","Pass");
// اغلاق الاتصال بقاعدة البيانات ..
mysql_close($link);
?>
```

هكذا نكون انتهينا من هذا الفصل وسندخل في الفصل الثاني حيث ستقوم بتصميم أول منتدى لك .

المحتويات

رقم الإيداع

ISBN



المركز الرئيسي : **11** شارع د/محمد رأفت – محطة الرمل –
الإسكندرية
تليفون وفاكس: 4838326 (03)(+2)
موبايل: 01001634294 (+2)

www.daralbraa.com
Email: info@daralbraa.com